



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA

PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA – PIBIC

**INICIATIVAS NA PRESERVAÇÃO DA
APARÊNCIA: RECONSTRUÇÃO DA GEOMETRIA
A PARTIR DE IMAGENS HDR**
Explorando métodos de descrição e correspondência de
pontos de interesse em imagens HDR

Área do conhecimento: Ciências Exatas
Subárea do conhecimento: Computação
Especialidade do conhecimento: Processamento de Imagens

Relatório Final
Período da bolsa: de agosto/2018 a julho/2019

Este projeto é desenvolvido com bolsa de iniciação científica

PIBIC/COPES

Orientadora: Beatriz Trinchão Andrade de Carvalho
Coorientador: Daniel Oliveira Dantas
Autor: Welerson Augusto Lino de Jesus Melo

Sumário

1	Introdução	3
2	Objetivos	4
3	Metodologia	4
3.1	Detectores	4
3.2	Máscara do Coeficiente de Variação	5
3.3	Implementação do algoritmo DetectorCoV	7
3.4	Base de Dados	8
3.5	Técnica de Divisão da Imagem por Área a partir da Iluminação	9
3.6	Teste de Seleção	11
3.7	CrITÉrios de Avaliação	14
4	Resultados e discussões	15
5	Conclusão	20
6	Perspectivas de futuros trabalhos	21
7	Referências Bibliográficas	21
8	Outras Atividades	22
9	Justificativa de Alteração no Plano de Trabalho	22

1 Introdução

O presente relatório traz os resultados obtidos no período de agosto de 2018 a julho de 2019 do Projeto de Pesquisa PVB6533-2018 - Iniciativas na preservação da aparência: reconstrução da geometria a partir de imagens HDR, "Explorando métodos de descrição e correspondência de pontos de interesse em imagens HDR".

Foi dada continuação ao plano "Modificando um algoritmo de obtenção de nuvem de pontos 3D para o suporte a imagens HDR" do Projeto de Pesquisa PVB5849-2017.

Muitas tarefas de visão computacional, tais como reconstrução 3D e *image stitching*, dependem da detecção de Pontos-Chave (PCs) [1, 2]. Em geral, algoritmos de geração de nuvens baseados em imagens têm como primeiro passo a detecção de PCs. Detectores de PCs usam imagens de Baixa Faixa Dinâmica - *Low Dynamic Range* (LDR) como entrada. Uma imagem LDR geralmente usa 8 bits por amostra. Desse modo, a amplitude da intensidade da luz armazenada num canal de cor de um pixel da imagem se limita ao intervalo $[0, 255]$. Por causa disso, métodos de aquisição de PCs baseados em imagens LDR processam indevidamente imagens onde existem cenas com uma grande variação na intensidade da luz, deixando áreas superexpostas muito claras ou áreas subexpostas muito escuras.

Como consequência, detectores de PCs podem falhar na detecção de pontos nestas áreas em imagens LDR. Imagens de Alta Faixa Dinâmica ou *High Dynamic Range* (HDR) geralmente usam mais de 8 bits por amostra, permitindo uma faixa dinâmica maior [3]. Como exemplo, o intervalo $[0, 2^{32} - 1]$ para uma imagem de 32 bits por canal de cor. Consequentemente, há uma maior precisão de cor nas áreas de superexposição e subexposição. Assumindo que os detectores de PCs são altamente dependentes da iluminação da cena no momento da captura [4], imagens HDR tendem a ser mais robustas que as imagens LDR na detecção desses pontos.

No plano anterior foram estudados, de forma mais aprofundada, os algoritmos de detecção de PCs. Os detectores estudados e implementados, seguindo os critérios de escolha descritos no plano, foram: Harris Corner Detector [5], Difference of Gaussian (DoG) [6] e o Speeded-Up Robust Features (SURF) [7]. Em uma terceira etapa, estudamos formas de aprimorar a detecção de PCs em imagens HDR, visto que a detecção neste tipo de imagem não possuía bons resultados [8]. Com isso, adicionamos na implementação dos três algoritmos citados uma técnica intermediária chamada de Máscara do Coeficiente de Variação (MCV) fazendo com que as imagens HDR se adaptassem melhor à detecção de PCs nesses algoritmos.

Os resultados obtidos nessa fase e publicados no artigo "Improving Feature Point Detection in High Dynamic Range Images" [9] mostraram que as alterações propostas nos algoritmos de detecção de PCs aplicados às imagens HDR através da MCV obtiveram, em geral, quando comparadas com os algoritmos originais, uma melhor resposta às métricas utilizadas.

Vislumbrando na aplicação da MCV em imagens HDR um possível ramo de estudo ainda não explorado de forma mais profunda, alguns experimentos iniciais foram realizados. De forma visual e empírica, notou-se, inicialmente, a semelhança entre as imagens de resposta da MCV e do passo derivativo do Harris Corner Detector. Desse momento em diante, um algoritmo totalmente novo para detecção de PCs em imagens HDR foi modelado e implementado com o objetivo de ser um algoritmo de detecção especialista em imagens HDR. O algoritmo tem por finalidade extrair as melhores características desse tipo de imagem. Chamamos esse algoritmo de DetectorCoV.

A grande maioria dos algoritmos detectores de PCs, inclusive os citados neste trabalho, possuem algum processo derivativo [8]. Esses processos derivativos se assemelham visu-

almente ao processo de aplicação da MCV em imagens HDR. Diante disso, foi formulada a hipótese de que: o processo derivativo na detecção de PCs em imagens HDR pode ser substituído pela MCV; além disso, a MCV pode melhorar, no sentido da repetibilidade e uniformidade dos pontos, a detecção neste tipo de imagens.

Foi selecionada uma técnica de Filtragem e um Fator Multiplicativo para que, combinadas com a MCV, completassem o algoritmo ajudando a melhorar a detecção. Duas novas bases de dados foram integralizadas para a realização e validação da técnica de Filtragem e do Fator Multiplicativo. Chamaremos esse teste de Teste de Seleção.

Após uma pré-seleção das técnicas, realizamos experimentos com banco de imagens controladas e avaliamos os resultados, selecionando as técnicas que melhor se adaptaram ao novo algoritmo. Além disso, alguns parâmetros relacionados à MCV foram testados para que pudéssemos selecionar a combinação de técnica e parâmetro que melhor se adequasse ao nosso objetivo que é melhorar a detecção de PCs em imagens HDR.

Nos dedicamos a fazer rotinas de experimentos para encontrar os parâmetros e técnicas que compõem o algoritmo. Além disso, para que os parâmetros pudessem ser realmente avaliados sem que houvesse vícios e os resultados tivessem maior consistência, nós fizemos a aquisição de outras bases de imagens. Por fim, pudemos comparar os resultados do novo algoritmo proposto com os resultados já existentes na literatura. Vimos que quando aplicado à imagens HDR, os resultados foram positivos.

2 Objetivos

Este plano de trabalho teve o objetivo de verificar a possibilidade de aprimoramento da detecção de pontos de interesse através da criação de um novo método que leva em consideração as especificidades das imagens HDR. Para isso, tivemos como objetivos específicos: projetar, implementar e validar a partir de dois critérios de avaliação existentes.

3 Metodologia

Nesta seção iremos detalhar as principais definições encontradas na literatura as quais este trabalho se baseou para poder criar o novo algoritmo de detecção de PCs em imagens HDR a partir da MCV. Também revisaremos a literatura dos detectores estudados anteriormente, a base de dados e métricas utilizadas para avaliar a detecção de PCs.

3.1 Detectores

A detecção de PCs numa imagem é fundamental em várias aplicações de Visão Computacional por ser o primeiro passo dos algoritmos dessas aplicações. Os detectores de PCs retornam os pixels de áreas consideradas relevantes na imagem. Os algoritmos de detecção selecionam os pixels (pontos) que podem ser considerados chaves (importantes) e estabelecem um valor para cada ponto. O valor estabelecido para o ponto será usado, através de algum tipo de limiar, para selecionar os PCs finais.

Os detectores de PCs estudados anteriormente, que aqui chamamos de detectores tradicionais, foram: Harris Corner Detector [5], algoritmo clássico de detecção e amplamente citado na literatura; Difference of Gaussian (DoG) [6], amplamente citado e usado em aplicações do mundo real; e o Speeded-Up Robust Features (SURF) [7], variação do SIFT a fim de melhorar a performance visando aplicações em tempo real.

Na literatura, poucos trabalhos avaliam a performance das imagens HDR na detecção de PCs [10, 8, 11, 12, 13]. Esses trabalhos mostram que imagens LDR baseadas em

HDR têm melhores resultados na detecção de PCs. Porém os resultados das imagens puramente HDR, que não passa por nenhuma transformação ou mapeamento de tons, não são satisfatórios, pois os algoritmos não exploram totalmente a capacidade desse tipo de imagens.

A grande maioria dos algoritmos detectores de PCs possui um passo intermediário derivativo. Os detectores assumem uma imagem LDR como entrada. Nas imagens LDR a magnitude das derivadas nas áreas escuras e claras são significativamente diferentes [8]. Essa diferença se torna mais evidente nas imagens HDR, pois elas possuem uma faixa dinâmica maior, fazendo com que a diferença entre os valores das partes claras e os valores das partes escuras das imagens seja maior. Os detectores de PCs usam algum tipo de limiar para filtrar os pontos mais fracos encontrados. Dessa forma, os pontos mais fortes são detectados em sua maior parte nas áreas mais claras da imagem, deixando os PCs das áreas escuras não detectados [8].

Por este motivo, Pribyl et al. [8] mostraram que os detectores processam as imagens HDR de forma ineficiente, não aproveitando todo o seu potencial.

No plano anterior foi proposto um aprimoramento dos três algoritmos acima para a detecção em imagens HDR. O aprimoramento usa a MCV como passo prévio aos passos derivativos. Essa abordagem foi chamada de Harris For HDR, DoG For HDR e SURF for HDR, referentes respectivamente ao Harris Corner Detector, DoG e SURF.

Todos os algoritmos citados nesta subseção foram detalhados no relatório final do projeto PVB5849-2017.

3.2 Máscara do Coeficiente de Variação

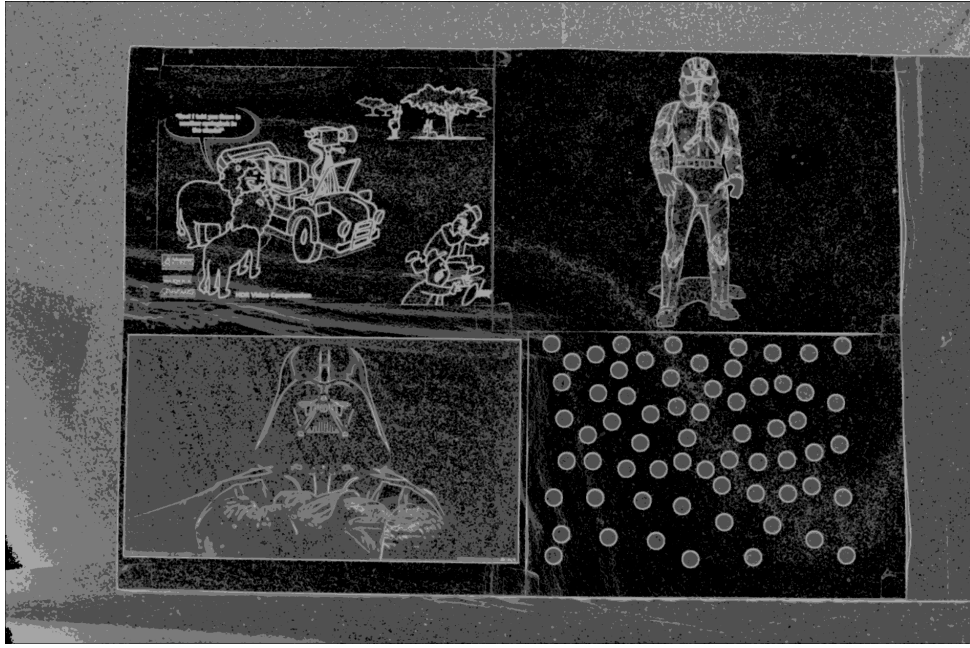
Para resolver o problema da detecção ineficiente nas imagens HDR, nós estudamos uma forma de normalizar as respostas das derivadas para que os seus valores nas áreas mais escuras estejam numa escala equivalente às respostas das derivadas nas áreas mais claras. Dessa forma, os pontos encontrados nessas áreas podem ser comparadas de forma justa na etapa de limiar. Para isso, estudamos a aplicação do método estatístico do Coeficiente de Variação (CV) em imagens.

Sabendo que estamos lidando com variações, ou seja, dispersões, em diferentes áreas da imagem, e o desvio padrão μ é medida de dispersão dos valores sobre uma população, podemos aplicar o CV como uma máscara local nas imagens. No CV, para diferentes populações ou fonte de dados, a média e o desvio padrão tendem a crescer ou diminuir juntos [14]. Logo, o CV é relativamente estável ou constante. Em adição, os valores calculados dentro de uma máscara na imagem serão afetados o mínimo possível por diferentes fontes de luz, já que é uma máscara local.

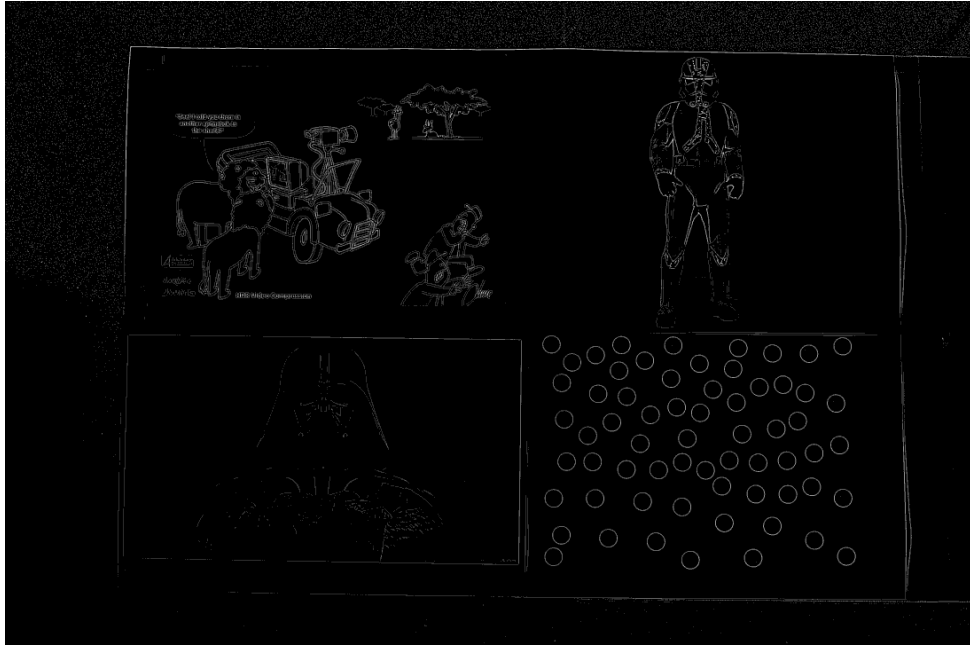
Em processamento de imagens, o CV é usado como um filtro para redução de ruídos em imagens como também para encontrar arestas [15, 16]. Filtros baseados no CV permitem a boa localização de arestas tanto em áreas claras quanto escuras de uma imagem [15]. Por essa razão, foi abordado no problema da detecção de PCs em imagens HDR a MCV como forma de melhorar a detecção nas áreas escuras deste tipo de imagem. A máscara local $n \times n$ do CV em imagens usa a Equação (1). Nossa população é o conjunto de $n \times n$ pixels na janela em volta de um pixel central (x, y) .

$$cv(p) = \frac{\sigma}{\mu}, \quad (1)$$

onde σ é o desvio padrão, definido na Equação 2.



(a)



(b)

Figura 1: Imagens intermediárias do processo de detecção de PCs do: (a) algoritmo DetectorCoV; (b) algoritmo Difference of Gaussian

$$\sigma = \sqrt{\frac{1}{N} \times \sum_{i=1}^N (p_i - \mu)^2}, \quad (2)$$

onde N é o número de pixels na máscara, isto é, o tamanho da população, p_i é o i -ésimo pixel dentro da máscara e μ a média aritmética da população.

3.3 Implementação do algoritmo DetectorCoV

A implementação do nosso algoritmo de detecção proposto para imagens HDR, que chamamos de DetectorCoV, foi baseada na MCV. A principal diferença, a priori, entre os algoritmos propostos no plano anterior (derivativa + MCV) e a nova abordagem somente com a MCV está no fato do novo algoritmo possuir uma complexidade de tempo menor. Porém, a complexidade final só poderá ser realmente avaliada ao final quando todos os parâmetros estiverem definidos. Uma outra motivação para a implementação deste novo algoritmo é a hipótese de que a detecção de PCs em imagens HDR tenha um resultado melhor ou igual aos algoritmos citados. Confirmada a configuração final, poderemos comparar os algoritmos em relação à complexidade como também a acurácia na detecção com base nos critérios de avaliação.

Antes de avançarmos, ressaltamos uma alteração que foi feita na fórmula do Coeficiente de Variação para que os pontos encontrados possam ficar mais centralizados em relação ao centro da máscara. Dessa forma, pretendemos melhorar a repetibilidade, um dos critérios de avaliação, como também esperamos uma leve mudança no conjunto dos pontos detectados como PCs. Para tal, a partir da equação do desvio padrão, Equação (2), adicionamos um peso w_i a cada subtração. O peso w_i representa uma distribuição Gaussiana 2D em uma região com N pixels em volta do i -ésimo pixel. A Gaussiana 2D possui um valor de desvio padrão próprio, que chamamos de σ_g . Ficamos com uma nova equação para o desvio padrão da MCV:

$$\sigma_2 = \sqrt{\frac{1}{N} \times \sum_{i=1}^N ((p_i - \mu)^2) w_i}, \quad (3)$$

onde N é o número de pixels da máscara, $p_i = f(x, y)$ tal que (x, y) está dentro da região da máscara na imagem é o valor do i -ésimo pixel dentro dessa região.

$$w_i = f(x, y) G(x, y) = f(x, y) \frac{1}{2\pi\sigma_g^2} e^{-\frac{x^2+y^2}{2\sigma_g^2}}, \quad (4)$$

onde x e y são as coordenadas da imagem dentro da região da máscara.

Por fim, a máscara do coeficiente de variação utilizada segue a Equação 5:

$$mcv(p) = \frac{\sigma_2}{\mu}, \quad (5)$$

Nesse plano de trabalho, a MCV foi transformada no passo fundamental para um novo algoritmo de detecção. Porém, como visto, a MCV é a princípio um detector de arestas. Dessa forma, alguns passos no algoritmo após a aplicação da MCV devem ser feitos para amenizar o efeito das arestas na detecção dos pontos. Chamaremos esses passos de Técnicas, as quais dividimos em dois tipos: Fator Multiplicativo e Filtro. Dentre as técnicas de fatores multiplicativos, estudamos a Transformação Linear, Equalização do Histograma e a Transformação Logarítmica [17]. Os filtros estudados foram: Filtro Gaussiano e Filtro Bilateral [17].

O primeiro passo exatamente após a aplicação da MCV é a *local maxima supression* em uma vizinhança 21×21 [9]. Esse passo seleciona o maior ponto dentro da vizinhança. Para trabalharmos com imagens HDR, uma normalização é feita para que a imagem fique dentro de um intervalo inteiro e controlável pois cada imagem HDR possui um intervalo de valores pré definido e geralmente desconhecido. Por causa desta variação e da característica de preservação de arestas da MCV, os pontos selecionados na etapa citada podem se concentrar em poucas áreas da imagem. Geralmente os pontos se concentram

em torno das arestas mais grossas, o que, consequentemente, é ruim para os critérios de avaliação. Devido a esse fato, a imagem de resposta da MCV foi tratada para reduzir os esses efeitos através de técnicas de transformação de histograma, etapa do algoritmo que nomeamos Fator Multiplicativo.

Dentre as técnicas utilizadas no Fator Multiplicativo, a primeira foi a multiplicação direta dos pixels da imagem intermediária por uma constante K , inicialmente definida como 25 (valor obtido de forma empírica). A segunda técnica foi a aplicação da Transformação Logarítmica, transformação representada pela fórmula $c \times \log(f(x, y))$ onde c é uma constante definida de forma empírica para 150 e $f(x, y)$ é uma imagem. A terceira foi a Equalização do Histograma. A definição de Equalização do histograma pode ser encontrada em Gonzales [17]. Os valores prefixados de forma empírica citados acima foram feitos a partir de testes com uma imagem de uma base de dados e a partir da observação do comportamento dos PCs encontrados. Desta forma, esses valores poderão ser alterados futuramente mediante experimentos após a conclusão dos experimentos correntes.

Após a aplicação de uma das técnicas apresentadas no parágrafo anterior, a imagem intermediária adquire alguns ruídos pontuais que podem ser decisivos para a detecção de bons PCs. Logo, como próximo passo, filtramos os ruídos gerados pelo Fator Multiplicativo através de duas técnicas bastante conhecidas. A primeira técnica é o Filtro Gaussiano, que utiliza a Equação (4) como base, causando um efeito de borramento na imagem dada. A segunda técnica é o Filtro Bilateral que possui características similares ao Filtro Gaussiano. O Filtro Bilateral também usa a Equação (4) como base, porém tentando preservar as características das arestas. As duas técnicas de filtragem de ruídos podem ser encontradas em Gonzales [17].

Nos passos finais do algoritmo, ao final do processo acima, a imagem intermediária é submetida ao processo de *local maxima supression* em uma vizinhança 21×21 . Como resultado, os pixels selecionados serão considerados como PCs da imagem. Porém, como forma de limiar, somente os 500 pontos mais fortes (os 500 primeiros em ordem decrescente) são usados.

Podemos então sumarizar o algoritmo DetectorCoV proposto nas seguintes etapas: Aplicação do Coeficiente de Variação com peso Gaussiano; Aplicação do Fator Multiplicativo na imagem intermediária; Aplicação do Filtro para Ruído; Extração dos PCs.

Na Figura 1(a) podemos ver um exemplo da imagem de resposta da MCV com o peso Gaussiano. Já na Figura 1(b), um exemplo da imagem de resposta da derivada do algoritmo DoG.

3.4 Base de Dados

As bases de dados são produzidas com o objetivo avaliar a estabilidade dos algoritmos de detecção ao lidar com alguns tipos de variações. As imagens da base são então geradas com algumas dessas variações. As variações mais comuns na literatura são: distância, iluminação e ponto de vista.

Para a realização dos experimentos nós utilizamos inicialmente a base de dados de imagens HDR e utilizada no plano e disponibilizada por Prybil et al. [8], denominada 2D. Porém, para os objetivos do plano atual, fez-se necessário ter uma variação maior de cenas para que os parâmetros sejam bem definidos e livres de padrões característicos de uma cena específica. Por essa razão, utilizamos também a base de dados 3D disponibilizada por Prybil et al. [8]. A Figura 2 mostra uma das cenas da base de dados 2D e a Figura 3 mostra uma das cenas da base de dados 3D. Essas duas bases foram utilizadas na etapa do Teste de Seleção que definiu os parâmetros do algoritmo DetectorCoV.

As cenas disponibilizadas pelo Prybil foram projetadas de forma que contenham três

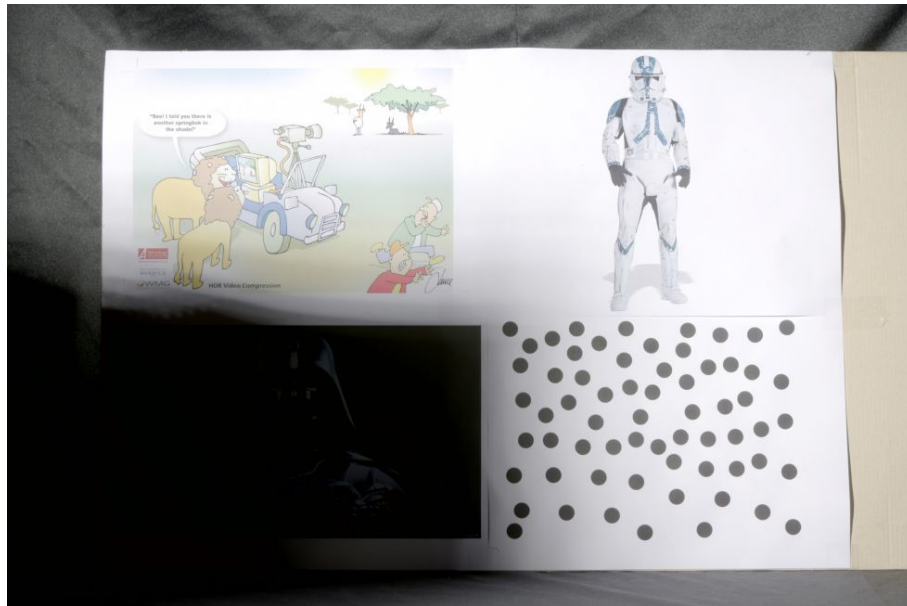


Figura 2: Imagem de uma cena da base de dados 2D

áreas não sobrepostas com diferentes intensidade de iluminação. Dessa forma, pode-se avaliar a distribuição dos PCs de acordo com a variação da iluminação. As áreas foram chamadas de: shadows (penumbra), midtones (tons médios) e highlights (tons claros). A técnica de avaliação Uniformity utiliza dessa divisão da cena em áreas para calcular o quão bem distribuídos estão os PCs encontrados. Ou seja, o algoritmo deve encontrar, de maneira uniforme, PCs nas áreas bem iluminadas, assim como, nas áreas escuras de uma cena.

As bases de dados 2D e 3D, foram utilizadas no Teste de Seleção para a obtenção dos parâmetros do algoritmo proposto. Dessa forma, uma maneira mais justa de comparar os resultados do algoritmo proposto com os já existentes é compará-los a partir de uma nova base de dados. Para isso, nós utilizamos a base de dados de imagens HDR ProjectRoom disponibilizada por Rana et al. [12]. A Figura 4 é uma exemplo de uma cena da base.

A base de dados ProjectRoom possui cenas 3D com vários objetos e possui variação de iluminação. A iluminação é controlada e em cada variação existe uma configuração de iluminação diferente na cena. Para mais detalhes sobre como a cena foi projetada, vide Rana et al. [12].

Como exposto acima, as bases de dados 2D e 3D possuem uma divisão prévia da cena em áreas de acordo com a iluminação. A base ProjectRoom, assim como as outras bases HDR que encontramos na literatura, não possui a divisão das cenas a partir da iluminação das áreas da imagem. Dessa forma, foi necessário fazer a divisão da base ProjectRoom. Após não encontrarmos nenhum modelo formal de como realizar essa divisão, nós elaboramos uma técnica automática de divisão da imagem por área de acordo com a iluminação. A técnica está descrita na Subseção 3.5.

3.5 Técnica de Divisão da Imagem por Área a partir da Iluminação

A técnica que desenvolvemos para a divisão da imagem de entrada HDR por área a partir da iluminação tem como objetivo dividir a imagem em duas áreas. Cada área, não sobreposta e não necessariamente conexa, deve ter características semelhantes de iluminação, ou seja, queremos uma área com pixels mais bem iluminados que a outra.



Figura 3: Imagem de uma cena da base de dados 3D

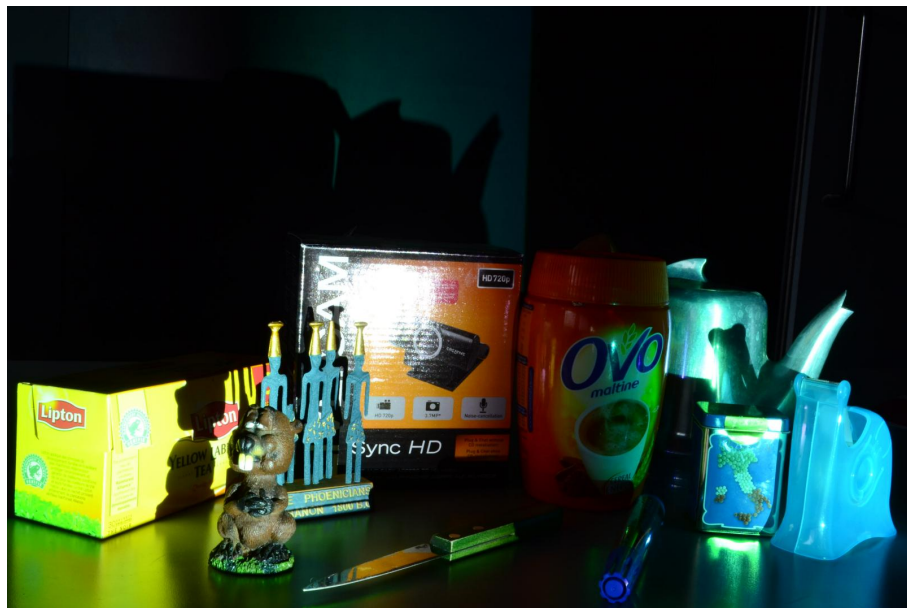


Figura 4: Imagem de uma cena da base de dados ProjectRoom

A escolha de uma boa técnica é fundamental para a avaliação realizada pelo método Uniformity.

Para isso, testamos três técnicas: usar a média do mapa luminância dos pixels da cena como limiar; usar a imagem média de todas as imagens da base de dados ProjectRoom; e usar metade dos pixels mais iluminados a partir da luminância. A última técnica foi a escolhida pois foi a que melhor representou, visualmente, a separação das áreas. Comparamos, por exemplo, a área visivelmente mais escura da imagem de entrada com a área que o método automático demarcou como escura. Além disso, levamos em consideração que as duas áreas possuísem objetos, ou parte de objetos, da cena para que houvesse potenciais PCs nas duas áreas.

Para dividir a imagem por área a partir da iluminação da cena, precisamos primeiramente ter alguma informação relativa à iluminação da cena em questão. Uma maneira de encontrar essa informação é obtendo a imagem de luminância da cena. Uma das formas de calcular a luminância em imagens HDR baseia-se no algoritmo de Retinex. No Retinex, a luminância é estimada a partir do mapa de radiância da imagem. Como Retinex é um problema matemático mal-posto segundo a definição de Hadamard, não há como obter a luminância exata de uma cena. De acordo com a teoria do Retinex, a imagem de entrada I é produto da luminância L da cena e da refletância R , i.e., $I = LR$.

Para encontrar L , nós iremos usar a abordagem Gaussiana aplicada no algoritmo em Chiu et al. [18]. Fazemos $L = I * G_\sigma$, onde G_σ é um filtro gaussiano e o parâmetro σ depende do tamanho da imagem $[m \times n]$, ou seja, $\sigma = \alpha \max(m, n)$. A constante alfa foi definida $\alpha = 0.007$ e o tamanho da máscara da gaussiana como o próximo número impar do inteiro obtido de 6σ . A imagem L é normalizada para 16 bits. Na Figura 5 podemos ver um exemplo da imagem de luminância L em cores falsas, calculada da imagem de entrada da Figura 4.

Na etapa seguinte, nós geramos o histograma acumulado H de L . Definimos um valor *mid* que será o limiar que divide a imagem em duas áreas: baixa iluminação e alta iluminação. O valor *mid* é referente à intensidade do pixel onde se encontra a metade do histograma acumulado H , podendo estar no intervalo $[0, 1023]$. Ou seja, as duas áreas têm a mesma quantidade de pixels, metade dos pixels da imagem serão definidos como baixa iluminação e a outra metade como alta iluminação. Vale ressaltar que desconsideramos desde o início do processo todos os pixels referente ao fundo da cena. Fizemos isso pois estamos interessados somente nos objetos da cena e o fundo é uma região que não possui objetos e detalhes que possam ser detectados como PCs. O fundo foi recortado manualmente e é o mesmo para todas as imagens da base supracitada.

Por fim, temos a divisão da imagem em duas áreas de iluminação distintas gerada de forma automática para cada imagem da base de dados ProjectRoom. Porém, esse método pode ser implantado para qualquer base de imagens HDR com uma alta variação de luz na cena. A divisão final da imagem HDR da Figura 4 pode ser vista na Figura 6.

3.6 Teste de Seleção

A partir do que foi exposto sobre o algoritmo DetectorCoV, vimos que algumas etapas do que chamamos de Técnica Fator Multiplicativo e Técnica de Filtragem, ou Filtro, possuem mais de uma opção. Além disso, alguns parâmetros internos às técnicas de filtragem interferem no resultado final do detector. Desse modo, precisamos fazer a seleção das técnicas e parâmetros a fim de encontrar a melhor combinação entre eles.

Os parâmetros e técnicas experimentados foram:

1. Fator Multiplicativo: as técnicas testadas foram: Transformação Linear, Transformação Logarítmica, Equalização do Histograma.

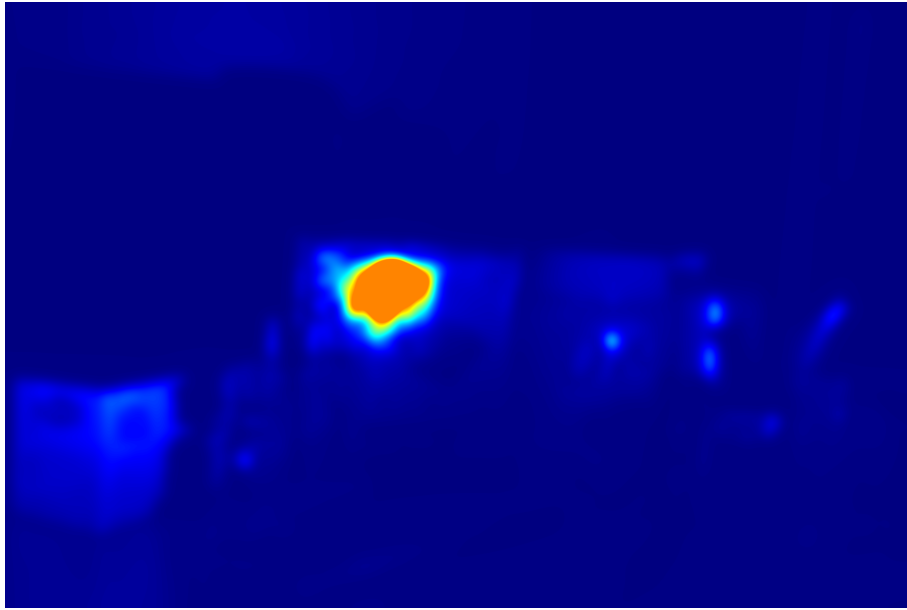


Figura 5: Mapa de Luminância gerado da imagem HDR da Figura 4 representado como mapa de calor onde as cores mais quentes representam regiões bem iluminadas.

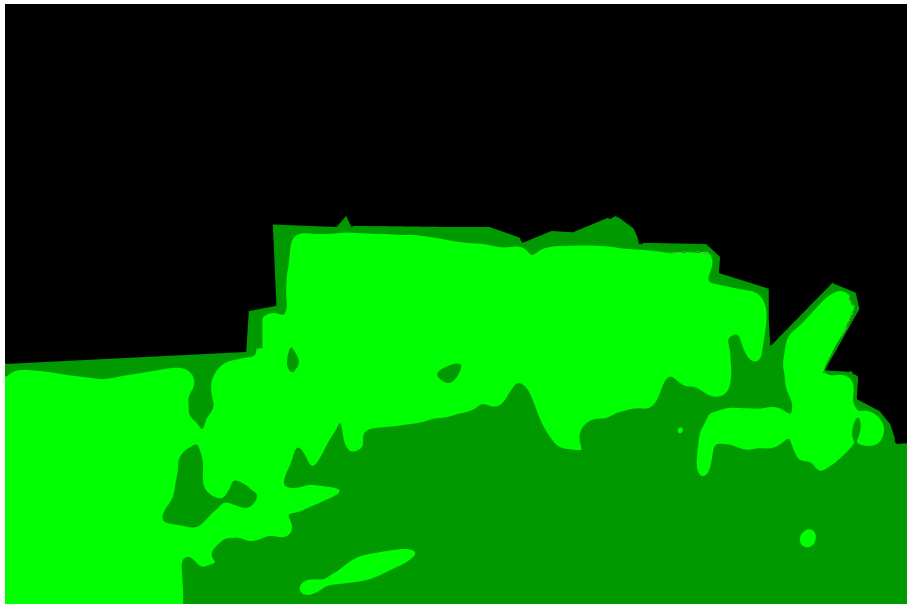


Figura 6: Divisão da imagem HDR da Figura 4, onde a cor preta representa o fundo da cena, a cor verde-escura a área de baixa iluminação e a cor verde clara a área de alta iluminação.

2. Filtro: Gaussiano, com os possíveis tamanho de máscaras: 5×5 , 9×9 , 15×15 ; Bilateral, com os possíveis valores de sigmas (para o $\text{sigmaSpace} = \text{sigmaColor}$): 150, 175, 200 (largura do filtro é fixada em 9 para todos os casos).

Isso nos dá um total de 18 combinações, que podem ser representadas em forma de conjunto como:

$$\{TLinear, TLogaritmica, EqHistograma\} \times (\{FiltroGaussiano\} \times \{5, 9, 15\} \cup \{FiltroBilateral\} \times \{150, 175, 200\}),$$

onde TLinear é a Transformação Linear, TLogaritmica a Transformação Logarítmica e EqHistograma a Equalização do Histograma.

Além disso, o desvio padrão σ_g da máscara w_i da Equação (4), pode variar entre os seguintes valores: 0.0, 1.0, 1.5, 2.0, 2.5. Assim, temos um total de 72 combinações, das quais somente uma será escolhida. Ainda houve a variação da base de dados, sendo as bases usadas neste teste de seleção a base 2D com as imagens do tipo HDR e logHDR e a base 3D com as imagens do tipo HDR e logHDR, ambas fornecidas por Pribyl et al. [8]. Deste modo, foram geradas 4 tabelas, cada uma com 72 tuplas de valores representando os dois critérios de avaliação usados, a Uniformity e a RR, que serão abordados na subseção seguinte. As tabelas geradas estão no Apêndice no fim deste relatório.

Determinadas as bases e os possíveis valores de parâmetro, foi feita uma rodada de experimentos para cada tipo de imagem nas bases de dados citadas. Em cada rodada o algoritmo DetectorCoV foi executado em todas as variações (distância, iluminação e ponto de vista) de imagens da cena da base de dados 2D, totalizando 36 imagens HDR e 36 imagens logHDR. Ao final do processo foi calculada a média da Uniformity e RR dessas imagens gerando-se a tupla (Uniformity, RR).

Já na base de dados 3D, foi utilizada somente a variação de iluminação, o que totaliza 7 imagens HDR e 7 imagens logHDR. Ao final do processo foi calculada a média da Uniformity e RR dessas imagens, também gerando-se a tupla (Uniformity, RR).

Com todos esses valores em mãos, precisávamos encontrar um valor que maximizasse os dois critérios de avaliação. Para isso o método de otimização de parâmetros e a relação de dominância entre vetores pôde ser aplicada. Considerando a tupla (Uniformity, RR) coordenadas de um vetor, a relação de dominância entre vetores é definida da seguinte maneira: o vetor $u = (u_1, \dots, u_k)$ domina um outro vetor $v = (v_1, \dots, v_k)$ se, e somente se, $u_i > v_i$ para todo i pertencente ao intervalo $[1, k]$. Uma visão aprofundada sobre o tema pode ser encontrada no livro texto *Evolutionary algorithms for solving multi-objective problems* [19].

Para a seleção do método, utilizamos as 18 possibilidades do conjunto:

$$\{TLinear, TLogaritmica, EqHistograma\} \times (\{FiltroGaussiano\} \times \{5, 9, 15\} \cup \{FiltroBilateral\} \times \{150, 175, 200\})$$

Cada uma das 18 configurações possíveis possui 4 tuplas, ou vetores, equivalentes à variação do desvio padrão da MCV. Foi calculada e somada a quantidade de vetores dominados por cada um dos 4 vetores para cada tipo de imagem de cada base de dados. Para escolher a combinação que obteve o melhor resultado, independente da base de dados e do tipo de imagem, nós somamos os valores de cada combinação para cada base de dados combinado com o tipo de imagem, como podemos ver na primeira coluna da Tabela 1. As duas últimas linhas da tabela equivalem ao somatório dos valores da coluna de acordo com o tipo de filtro, sendo a penúltima linha o somatório dos filtros Gaussianos de cada coluna e a última linha o somatório dos filtros Bilaterais de cada coluna.

Na Tabela 1, para facilitar a visualização, abreviamos o Fator Multiplicativo na primeira linha e o nome dos Filtros na segunda coluna.

Dataset-Tipo	Filtro	Hist1	Hist2	Hist3	Log1	Log2	Log3	Lin1	Lin2	Lin3
3D-HDR	Gauss	3	30	50	14	31	19	35	50	66
3D-HDR	Bilateral	42	51	75	29	39	32	16	16	34
3D-logHDR	Gauss	153	102	21	8	23	4	20	15	1
3D-logHDR	Bilateral	106	72	53	15	15	10	44	26	8
2D-HDR	Gauss	52	60	49	10	107	100	1	24	58
2D-HDR	Bilateral	77	95	125	133	134	144	40	56	61
2D-logHDR	Gauss	28	110	45	83	104	9	6	19	126
2D-logHDR	Bilateral	3	4	0	7	15	13	6	10	21
Somatório	Gauss	236	302	165	115	265	132	62	108	251
Somatório	Bilateral	228	222	253	184	203	199	106	108	124

Tabela 1: Cada célula da tabela contém o número de dominantes que um conjunto de pontos possui.

As associações são:

1. Gauss -> Filtro Gaussiano;
2. Bilateral -> Filtro Bilateral;
3. HistX -> Equalização do Histograma;
4. LogX -> Transformação Logarítmica;
5. LinX -> Transformação Linear;

onde X representa um número no intervalo [1,3]. O número após a abreviação representa uma das variações do Filtro combinada com o Fator Multiplicativo. Dessa forma, a combinação Gauss-F1, Gauss-F2 e Gauss-F3 equivalem, respectivamente, ao Filtro Gaussiano de tamanho 5×5 , 9×9 e 15×15 . Já a combinação Bilateral-F1, Bilateral-F2, e Bilateral-F3 equivalem, respectivamente, ao Filtro Bilateral com os sigmas iguais a 150, 175 e 200, onde F equivale ao Fator Multiplicativo.

A partir do somatório, pudemos escolher qual a combinação que mais dominou entre todas as possíveis. A combinação Gauss-Hist2, obteve um total de 302 vetores dominados, sendo portando a configuração ótima escolhida: o Fator Multiplicativo Equalização do Histograma e o Filtro Gaussiano de tamanho 9×9 .

Das ferramentas tecnológicas utilizadas, tivemos como principais ferramentas para a implementação dos algoritmos a linguagem de programação C++11 somada à biblioteca de Visão Computacional OpenCV 3.3.0. Além disso, foi desenvolvido para a realização dos experimentos alguns scripts em Python3 e Bash para Linux. Os experimentos foram executados em máquinas virtuais Linux do *Google Cloud Platform*.

3.7 Critérios de Avaliação

Em relação aos critérios de avaliação utilizados para comparar os algoritmos de detecção de acordo com os pontos encontrados, nós estudamos *Repeatability Rate* (RR), proposta por Schmid [1] e vastamente utilizada na literatura. O RR avalia a estabilidade de um detector a partir da repetição de pontos detectados em diferentes imagens da mesma cena, i.e., com algum tipo de variação. A RR está no intervalo [0.0, 1.0], onde o melhor resultado é 1.0. Isso significa que quanto maior a taxa RR, mais PCs correlatos foram encontrados em variações da mesma cena.



Figura 7: Exemplo da divisão feita automaticamente. As imagens da esquerda são as imagens de entrada. As imagens da direita são as imagens de saída com a área de baixa iluminação recortada da imagem, representada pela cor preta. As duas imagens de entrada são imagens da base de dados ProjectRoom com cenas que variam a iluminação.

Também utilizamos a Uniformity que propomos no plano anterior e publicamos no artigo "Improving Feature Point Detection in High Dynamic Range Images"[9]. A Uniformity avalia o quão bem distribuídos os PCs encontrados por um detector estão em uma imagem a partir de áreas pré estabelecidas. Essas áreas dividem a imagem em regiões claras, escuras e meio tons, sendo este último opcional. A Uniformity está no intervalo $[0.0, 1.0]$ sendo 1.0 o melhor resultado possível significando que os pontos chave estão igualmente distribuídos entre todas as áreas de diferente iluminação da cena.

Com os dois critérios acima, podemos avaliar tanto o quão estável é um algoritmo de detecção em relação às variações da cena, sendo elas variação de: distância, ponto de vista e iluminação. Como também o quão estável é esse mesmo algoritmo em relação à distribuição dos PCs na imagem.

Para mais detalhes de como os dois critérios de avaliação funcionam e como utilizamos-os para gerar os resultados, o relatório do plano de projeto passado ou os artigos citados nesta seção podem ser consultados.

4 Resultados e discussões

O primeiro resultado importante deste plano de trabalho foi a criação de uma técnica de divisão da imagem em áreas a partir da iluminação da cena. Esta técnica pode ajudar a resolver de maneira mais formal e automática o mesmo problema que encontramos na base de dados ProjectRoom caso outras bases sejam usadas. Os resultados da divisão da imagem foram avaliados visualmente a partir da análise das áreas geradas e a iluminação nessas áreas. Também avaliamos a quantidade de objetos ou parte de objetos na área, pois os objetos na cena são as principais fontes de PCs.

A Figura 7 exemplifica a divisão realizada em 2 cenas. Nas imagens da direita, todos os pixels considerados da região clara, alta iluminação, receberam a cor preta. Assim, a região que sobra na imagem é a região classificada como escura, baixa iluminação. Lembrando que o fundo da cena é desconsiderado na divisão. Fazendo isso para todas as imagens da base de dados, pudemos verificar que a técnica atingiu o objetivo que propomos.

O segundo resultado importante deste plano de trabalho, iniciou-se com a escolha do Fator Multiplicativo e do Filtro, respectivamente: Equalização do Histograma e o Filtro Gaussiano de tamanho 9×9 . Pudemos finalizar a estrutura do algoritmo DetectorCoV, e o resultado final do algoritmo está descrito de forma resumida logo abaixo:

Algoritmo 1: DETECTORCoV

Entrada: *Img*

Saída: Array de pontos (os pontos chaves)

```

1 início
2   responseImg  $\leftarrow$  MCV(Img)
3   responseImg2  $\leftarrow$  equalizacaoHistograma(responseImg)
4   responseImg3  $\leftarrow$  filtroGaussiano(responseImg2)
5   arrayPontos  $\leftarrow$  nonMaximaSupression(responseImg3)
6   ordeneDecrescente(arrayPontos)
7   maxPCs  $\leftarrow$  500
8   para i  $\leftarrow$  0 < maxPCs faça
9     | PCs.adicione(arrayPontos[i])
10  fim
11 fim
12 retorna PCs
```

Na Figura 8 podemos ver um exemplo dos PCs encontrados pelo algoritmo DetectorCoV. Os PCs estão representados por pequenos círculos vermelhos na borda preenchidos de verde. Podemos notar a quantidade grande de pontos encontrados em regiões mais escuras da imagem. A Figura 9 mostra o resultado da detecção do SURF, com a mesma quantidade de PCs encontrados na mesma cena da Figura 8. Podemos notar que os pontos encontrados estão concentrados em regiões mais claras da imagem, seguindo o padrão encontrado em cenas de outras bases de trabalhos anteriores [12, 9]. Vale lembrar que como essas imagens são transformações diretas das imagens originalmente HDR para uma imagem LDR com a extensão .png. Por esse motivo, a região mais clara da imagem está superexposta e não conseguimos notar os detalhes contidos nela.



Figura 8: PCs encontrados após aplicação do algoritmo DetectorCoV na imagem da base ProjectRoom

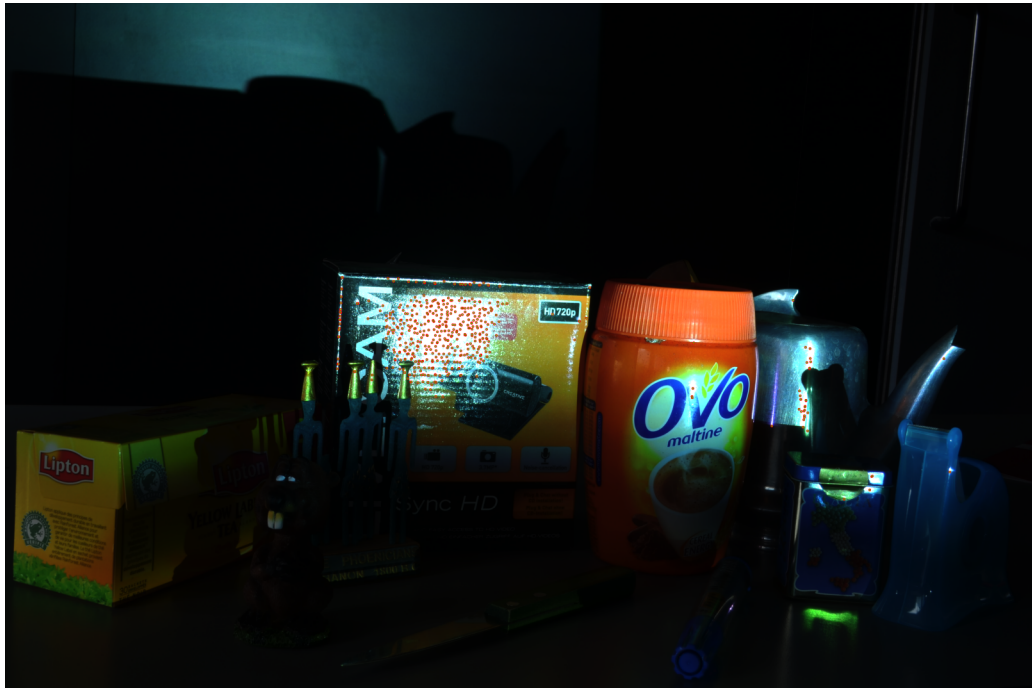


Figura 9: PCs encontrados após aplicação do algoritmo SURF na imagem da base ProjectRoom

Em seguida, veremos os resultados de RR e Uniformity do DetectorCoV aplicado à base ProjectRoom e o comparativo com 6 algoritmos de detecção aplicados à mesma base. Os 6 algoritmos foram citados na subseção Detectores da seção de Metodologia.

Os primeiros resultados são referentes à Uniformity. O DetectorCoV obteve 0.77 de no máximo 1.0. Para os algoritmos tradicionais os resultados foram 0.04, 0.02 e 0.00 referentes, respectivamente, ao Harris Corner, DoG e SURF. As baixas taxas de Uniformity,

ou seja, baixa distribuição dos pontos nas áreas claras e escuras da imagem, usando os algoritmos tradicionais já eram esperadas, visto os resultados obtidos anteriormente com outras bases, como mostramos em [9]. Já para os algoritmos Harris For HDR, DoG For HDR e SURF For HDR, propostos no plano anterior, obtivemos, respectivamente, 0.70, 0.70 e 0.23.

Os valores citados acima estão todos representados no gráfico da Figura 10. Como podemos ver no gráfico, o resultado alcançado com o DetectorCoV superou todos os outros algoritmos, estando com valor próximo ao dos algoritmos Harris For HDR e DoG For HDR.

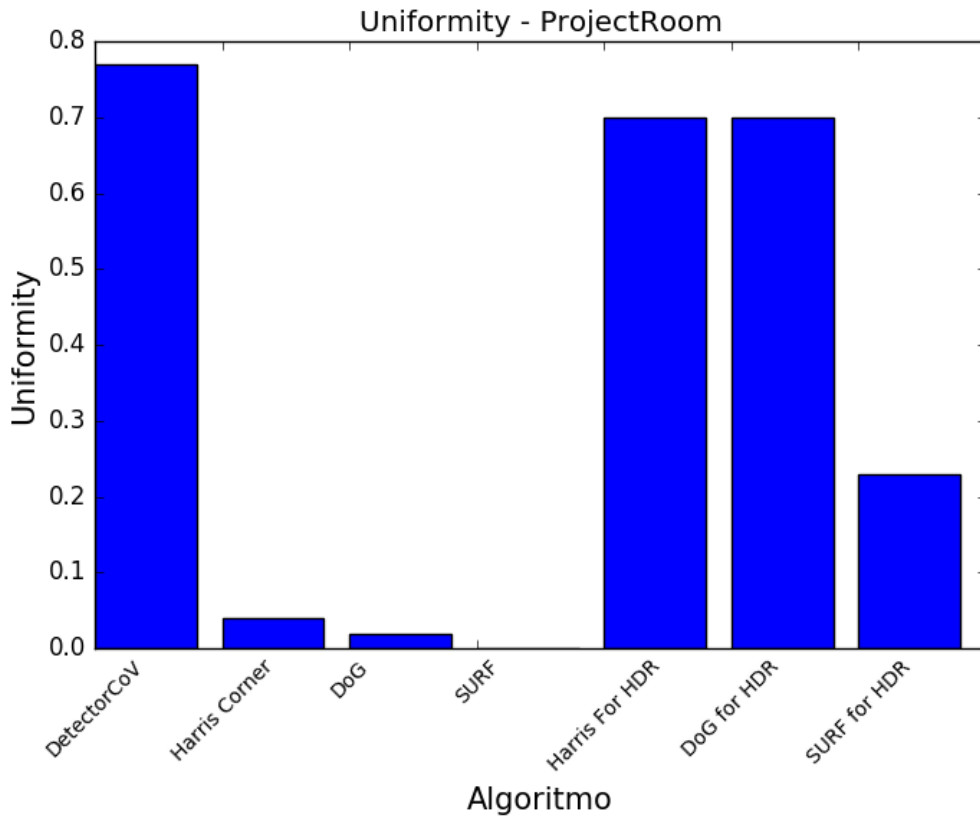


Figura 10: Resultados Uniformity na base de dados ProjectRoom

Nos resultados referentes à taxa de repetibilidade, a RR, o algoritmo DetectorCoV obteve 0.21 de no máximo 1.0. Para os algoritmos tradicionais os resultados foram 0.12, 0.14 e 0.33 referentes, respectivamente, ao Harris Corner, DoG e SURF. Já o Harris For HDR, DoG For HDR e SURF For HDR obtiveram, respectivamente, 0.28, 0.16 e 0.14.

No gráfico da Figura 12 podemos ver os valores acima lado a lado. Podemos observar que o DetectorCoV supera dois dos algoritmos tradicionais, Harris Corner e DoG, assim como dois dos novos algoritmos para HDR, o DoG for HDR e o SURF for HDR. O SURF tradicional obtém a maior RR entre todos, seguido pelo Harris For HDR.

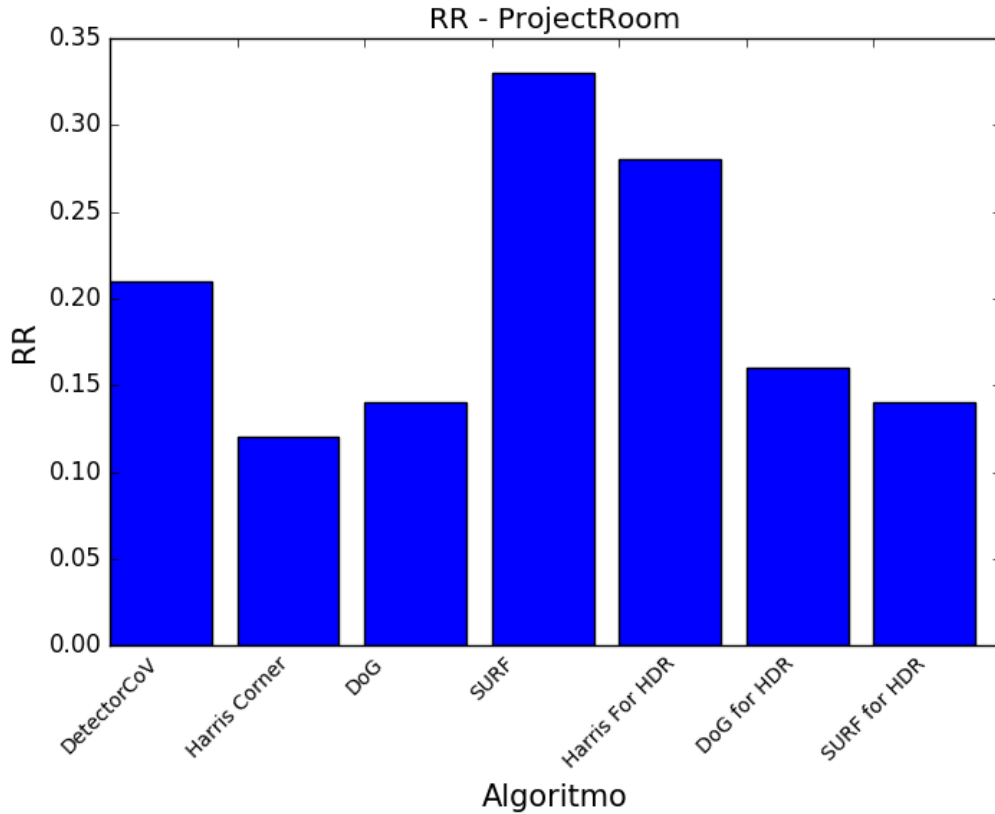


Figura 11: Resultados RR na base de dados ProjectRoom

Analisando os resultados de forma conjunta, vemos que apesar de uma taxa de repetibilidade alta, o algoritmo SURF possui uma baixa taxa de Uniformity. Isso mostra que apesar de grande parte dos pontos encontrados se repetirem em cenas distintas, esses pontos estão em sua maioria quase absoluta concentrados nas regiões mais claras da imagem. Já Harris For HDR teve um bom resultado nos dois critérios de avaliação. O resultado expressivo do Harris For HDR obteve desempenho superior ao esperado devido à sua estabilidade. Podemos explicar este resultado usando o fato da cena ser uma cena sem variação de distância e de ponto de vista, o que é bom para o Harris Corner que não é invariante a escala, nem de ponto de vista (transformações afins).

Como já abordado neste relatório, estudamos uma forma de analisar melhor a relação conjunta da Uniformity e RR a partir da relação de dominância entre vetores. Na Figura 12 foram distribuídos os pontos dos vetores com as coordenadas (Uniformity, RR) referente a cada algoritmo aplicado à base ProjectRoom.

A partir da definição de dominância, podemos observar diretamente no gráfico quantas pontos um certo ponto domina. Vemos que o ponto relativo ao SURF, o DoG e o Harris Corner não dominam nenhum outro ponto; o SURF For HDR domina 1 ponto; o DoG For HDR e o Harris For HDR dominam 3 pontos cada; e o DetectorCoV domina 4 pontos. Além disso, o SURF, o Harris For HDR e o DetectorCoV não são dominados por nenhum outro ponto. Dessa forma eles fazem parte de um subconjunto que pode ser utilizados como o subconjunto ótimo, ou seja, não foi superado por nenhum outro algoritmo.

Isso mostra que, de forma geral, se considerarmos o conjunto das duas formas de avaliação usadas para um algoritmo de detecção, o DetectorCoV obteve um excelente resultado dominando a maior quantidade de outros algoritmos e estando dentro do subconjunto ótimo.

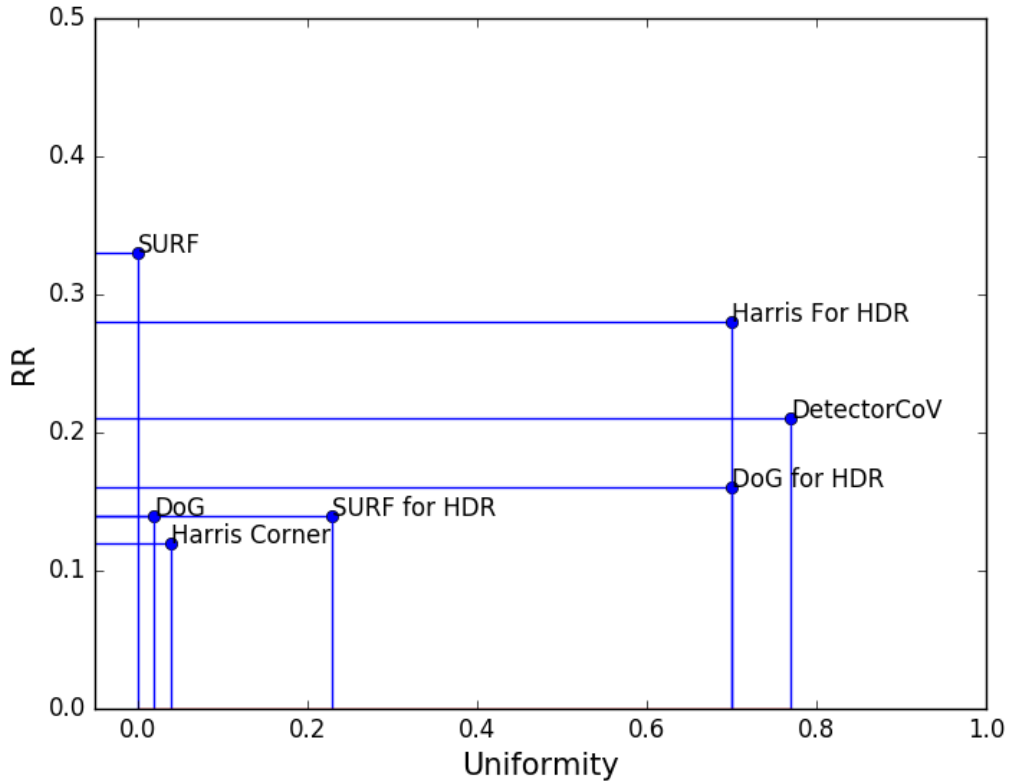


Figura 12: Projeção dos pontos (Uniformity, RR) para cada algoritmo na base ProjectRoom

5 Conclusão

Neste projeto de iniciação científica, pudemos estudar de forma um pouco mais aprofundada os algoritmos detectores de PCs, de forma especial a sua aplicação em imagens de alta faixa dinâmica, as imagens HDR. Vimos que os algoritmos detectores de PCs passam por algum processo derivativo. O processo derivativo é um passo fundamental para detecção, mas que também é o principal causador dos resultados ruins da detecção em imagens HDR. Nós percebemos a semelhança da imagem de saída desse processo com uma técnica aplicada por nós anteriormente, a Máscara do Coeficiente de Variação, MCV. A partir de então, foi investigado se era possível substituir o processo derivativo dos detectores pela MCV, e se a partir dessa substituição, as taxas de RR e Uniformity, as formas de avaliação utilizadas, permaneceriam ou melhorariam.

Vimos que com ajustes, ou seja, técnicas complementares à MCV, pudemos criar um algoritmo de detecção de pontos novo, substituindo o processo derivativo. Algoritmo esse que chamamos de DetectorCoV, no qual possui, visualmente, resultados bastantes semelhantes ao algoritmo DoG por ter respostas altas para arestas.

Para validar o algoritmo em relação aos critérios de avaliação RR e Uniformity, a base de dados ProjectRoom foi incluída. Os resultados obtidos na taxa de Uniformity mostrou que o algoritmo cumpre o papel de encontrar PCs em todas as áreas de iluminação da cena, o que era um fator importante se tratando de detecção em imagens HDR. No entanto, a taxa de RR não pode ser baixa pois descaracterizaria o algoritmo, mostrando que ele não possui bons PCs, apesar de ter encontrado PCs bem distribuídos. A taxa de RR para o DetectorCoV foi estável se comparada com a média de resultado dos outros algoritmos.

Podemos, por fim, concluir que o algoritmo proposto, DetectorCoV, mostrou-se viável para a detecção de PCs em imagens HDR. Obtivemos resultados com PCs razoavelmente estáveis às variações de iluminação contidas nas cenas da base ProjectRoom, e muito bem distribuídos entre as áreas de diferente iluminação da imagem.

6 Perspectivas de futuros trabalhos

Há perspectiva em aprofundar os estudos dos detectores em imagens HDR a partir de outros estudos futuros que surgirem na literatura, visto que este é um ramo ainda pouco explorado. Também pode-se estudar de forma mais profunda os algoritmos de geração de imagens HDR. O estudo da geração desse tipo de imagens pode fazer com que um novo algoritmo de geração de HDR, ou o aprimoramento de algum já existente, melhore a detecção de PCs em imagens HDR.

Além disso, podemos estender os estudos para os descritores de PCs, verificando quão significativa é a melhora da descrição e correspondência dos PCs encontrados com o algoritmo DetectorCoV e os outros algoritmos propostos no plano anterior. Por fim, poderemos gerar a nuvem de pontos 3D e avaliar as reconstruções geradas por uma imagem HDR, por uma imagem LDR e por uma imagem HDR utilizando algoritmos de detecção específicos para HDR.

7 Referências Bibliográficas

- [1] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of computer vision*, 37(2):151–172, 2000.
- [2] Richard Szeliski. *Computer Vision - Algorithms and Applications*. Texts in Computer Science. Springer, 2011.
- [3] F Banterle, A Artusi, K Debattista, and A Chalmers. Advanced high dynamic range imaging: Theory and practice. ak peters.
- [4] Krystian Mikolajczyk, Tinne Tuytelaars, and Cordelia et al. Schmid. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005.
- [5] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Manchester, UK, 1988.
- [6] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [7] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.
- [8] Bronislav Pribyl, Alan Chalmers, Pavel Zemčík, Lucy Hooberman, and Martin Čadík. Evaluation of feature point detection in high dynamic range imagery. *Journal of Visual Communication and Image Representation*, 38:141–160, 2016.
- [9] Welerson Augusto Lino Jesus de Melo, Jusley Arley Oliveira de Tavares, Daniel Oliveira Dantas, and Beatriz Trinchao Andrade. Improving feature point detection in high dynamic range images. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 00091–00096. IEEE, 2018.

- [10] Bronislav Přebyl, Alan Chalmers, and Pavel Zemčík. Feature point detection under extreme lighting conditions. In *Proceedings of the 28th Spring Conference on Computer Graphics*, pages 143–150. ACM, 2013.
- [11] Aakanksha Rana, Giuseppe Valenzise, and Frederic Dufaux. Learning-based tone mapping operator for image matching. In *IEEE International Conference on Image Processing (ICIP'2017)*, 2017.
- [12] Aakanksha Rana, Giuseppe Valenzise, and Frederic Dufaux. Optimizing tone mapping operators for keypoint detection under illumination changes. In *Multimedia Signal Processing (MMSP), 2016 IEEE 18th International Workshop on*, pages 1–6. IEEE, 2016.
- [13] G Kontogianni, EK Stathopoulou, A Georgopoulos, and A Doulamis. HDR imaging for feature detection on detailed architectural scenes. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(5):325, 2015.
- [14] George W Snedecor and William G Cochran. Statistical methods. ames, 1967.
- [15] Yongjian Yu and Scott T Acton. Edge detection in ultrasound imagery using the instantaneous coefficient of variation. *IEEE Transactions on Image Processing*, 13(12):1640–1655, 2004.
- [16] M Mora, C Tauber, and H Batatia. Robust level set for heart cavities detection in ultrasound images. In *Computers in Cardiology, 2005*, pages 235–238. IEEE, 2005.
- [17] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Pearson/Prentice Hall, 2008.
- [18] Ken Chiu, Michael Herf, Peter Shirley, S Swamy, Changyaw Wang, Kurt Zimmerman, et al. Spatially nonuniform scaling functions for high contrast images. In *Graphics Interface*, pages 245–245. CANADIAN INFORMATION PROCESSING SOCIETY, 1993.
- [19] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.

8 Outras Atividades

O bolsista participou do curso "Inglês para fins específicos: Tecnologia" com carga horária de 48 horas, Número do Registro 546497, com nota 8.80, oferecido pelo Núcleo de Línguas da Universidade Federal de Sergipe no âmbito do Programa Idiomas sem Fronteiras - Inglês, no período de 13/08/2018 a 01/11/2018. Participou do evento EV230-2018 - Persuasive Prose: Cover Letters, Essays, and Formal Writing in English. Participou também do minicurso CR394-2018 - V SEMAC - 28 EIC/COPES - REDAÇÃO CIENTÍFICA T4 (TARDE) oferecido pela SEMAC.

9 Justificativa de Alteração no Plano de Trabalho

O plano de trabalho do projeto atual tinha como objetivo inicial o estudo dos algoritmos de descrição de pontos de interesse, os PCs. No plano de trabalho anterior,

PVB5849-2017, nós visualizamos um caminho promissor no estudo da MCV, que resultou na publicação de um artigo nos proceedings do evento IEEE International Symposium on Computers and Communications. Desse modo, ao final do plano, alguns experimentos foram realizados tendo-se um indício de que usar a MCV no lugar de uma abordagem derivativa, como comumente é feito no algoritmo de detecção de PCs, leva a resultados semelhantes, a primeira vista, aos resultados encontrados com os algoritmos que usam a abordagem derivativa.

No entanto, a abordagem usando a MCV depende de alguns parâmetros, o que faz com que os resultados que obtivemos a princípio sejam melhorados ao passo que os parâmetros mais adequados sejam encontrados. Por esse motivo, no início do plano atual, nos dedicamos a fazer rotinas de experimentos para que esses parâmetros fossem encontrados. Além disso, para que os parâmetros pudessem ser realmente avaliados sem que houvesse interferência (vício) gerada pela base de imagens atual, nós fizemos a aquisição de outras bases para que os melhores parâmetros encontrados pudessem ser confirmados como os melhores em qualquer conjunto de imagens.

Com isso, o plano de trabalho foi alterado para podermos realizar os experimentos expostos acima e poder comparar a nova abordagem criada, ou seja, o novo algoritmo de detecção de PCs que desenvolvemos, com os já existentes, a fim de ter conhecimento da capacidade de detecção de pontos desse novo algoritmo proposto. No fim desse processo poderemos sugerir qual o melhor algoritmo de detecção de PCs combinados com imagens HDR e assim iniciar os estudos da descrição dos PCs neste tipo de imagens.

Apêndice A. Tabelas com resultados de Uniformity e RR

Dataset 2D prbyl et al.		Fator Multiplicativo = 25					Fator Multiplicativo = histEqualization					Fator Multiplicativo = 150*log(t(x,y))				
HDR	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5
SCV X GS	5 X 5	9 X 9	15 X 15	BL(10, 150)	BL(10, 175)	BL(10, 200)	5 X 5	9 X 9	15 X 15	BL(10, 150)	BL(10, 175)	BL(10, 200)	5 X 5	9 X 9	15 X 15	BL(10, 150)
0.0	(0.35; 0.20)	(0.54; 0.33)	(0.55; 0.41)	(0.52; 0.36)	(0.52; 0.42)	(0.51; 0.47)	(0.54; 0.36)	(0.53; 0.44)	(0.57; 0.39)	(0.59; 0.40)	(0.56; 0.44)	(0.59; 0.48)	(0.51; 0.33)	(0.56; 0.43)	(0.71; 0.41)	(0.60; 0.41)
1.0	(0.37; 0.26)	(0.55; 0.33)	(0.60; 0.35)	(0.52; 0.33)	(0.52; 0.35)	(0.53; 0.37)	(0.57; 0.36)	(0.62; 0.35)	(0.66; 0.36)	(0.64; 0.38)	(0.62; 0.40)	(0.65; 0.41)	(0.49; 0.26)	(0.58; 0.35)	(0.68; 0.38)	(0.64; 0.38)
1.5	(0.34; 0.34)	(0.55; 0.34)	(0.67; 0.35)	(0.58; 0.32)	(0.56; 0.34)	(0.57; 0.37)	(0.64; 0.36)	(0.69; 0.36)	(0.72; 0.30)	(0.71; 0.35)	(0.72; 0.38)	(0.73; 0.39)	(0.49; 0.25)	(0.64; 0.37)	(0.70; 0.32)	(0.69; 0.40)
2.0	(0.35; 0.34)	(0.53; 0.31)	(0.67; 0.32)	(0.67; 0.36)	(0.67; 0.38)	(0.66; 0.38)	(0.61; 0.33)	(0.70; 0.31)	(0.66; 0.27)	(0.66; 0.30)	(0.70; 0.33)	(0.72; 0.33)	(0.54; 0.28)	(0.71; 0.41)	(0.66; 0.29)	(0.74; 0.38)
LogHDR		CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5	CV Size = 5
SCV X GS	5 X 5	9 X 9	15 X 15	BL(10, 150)	BL(10, 175)	BL(10, 200)	5 X 5	9 X 9	15 X 15	BL(10, 150)	BL(10, 175)	BL(10, 200)	5 X 5	9 X 9	15 X 15	BL(10, 150)
0.0	(0.49; 0.41)	(0.50; 0.50)	(0.53; 0.53)	(0.48; 0.47)	(0.48; 0.51)	(0.48; 0.51)	(0.54; 0.49)	(0.53; 0.55)	(0.58; 0.49)	(0.48; 0.48)	(0.47; 0.52)	(0.45; 0.54)	(0.50; 0.55)	(0.54; 0.53)	(0.60; 0.46)	(0.48; 0.49)
1.0	(0.50; 0.45)	(0.48; 0.50)	(0.51; 0.54)	(0.48; 0.49)	(0.48; 0.49)	(0.48; 0.51)	(0.59; 0.46)	(0.55; 0.52)	(0.58; 0.48)	(0.47; 0.48)	(0.46; 0.52)	(0.44; 0.55)	(0.51; 0.52)	(0.56; 0.51)	(0.60; 0.40)	(0.50; 0.48)
1.5	(0.50; 0.45)	(0.48; 0.51)	(0.52; 0.53)	(0.50; 0.48)	(0.50; 0.49)	(0.51; 0.50)	(0.61; 0.42)	(0.59; 0.49)	(0.59; 0.45)	(0.51; 0.46)	(0.48; 0.50)	(0.45; 0.52)	(0.54; 0.50)	(0.58; 0.49)	(0.61; 0.38)	(0.53; 0.46)
2.0	(0.52; 0.47)	(0.49; 0.50)	(0.54; 0.51)	(0.53; 0.47)	(0.53; 0.47)	(0.53; 0.47)	(0.60; 0.39)	(0.62; 0.45)	(0.62; 0.42)	(0.54; 0.43)	(0.51; 0.47)	(0.47; 0.49)	(0.55; 0.48)	(0.60; 0.46)	(0.58; 0.35)	(0.56; 0.44)

Figura 13: Tabela (Uniformity, RR) Base de Dados 2D[8].

